Homework Assignment
CS 422, Spring 2010
Due: 4/7/10 by 5pm to staff mailing list

Overview

In this (one and only) homework assignment for the class you will be extending the "stalker" bot demoed on the first day of class to add some additional functionality. We would like everyone to complete this assignment individually, so you become comfortable with the software environment. You are welcome to discuss with your classmates, but should not look at each others' code. Please start early! Getting the environment set up will probably take 2-3 longer than the actual assignment and you may hit bumps along the way. The assignment should take no more than 100 lines of code. The goal of this assignment is two-fold:
- Make sure you have all the necessary software installed and working in preparation for the rest of the class.
- Give you a first taste of the Lib Openmetaverse API.

Prerequisites

Before doing anything, make sure you have all the software working as covered in the first two days of classes. This includes:
- Signed up for a SL account
- Attached the avatar shadow to your Intelligent Avatar and made sure it is working (i.e. writing to the database)
- Have access to ai.stanford.edu
- Installed Lib Open Metaverse.
- Have Visual Studio 2008 installed and tried compiling+running a "Hello World" C# program
- Installed MySQL Connector/ODBC and verified it connects successfully after opening an SSH tunnel to ai.stanford.edu

Getting Started

Start by downloading the 'CS422API' Visual Studio Solution from the class website. You should be able to open this in Visual Studio on your local machine and it should compile as is. Depending on where you installed the Lib Open Metaverse libraries, you may need to re-add OpenMetaverse.dll and OpenMetaverseTypes.dll as references to the solution. Assuming everything is set up properly, the API demos should also work; verify that it is working before proceeding with the rest of the assignment.

The stalker bot code is located in Follow.cs. Create another dummy SL account to follow your real Intelligent Avatar (IA). (I will be able to see the password of the stalker in plaintext, so don't use the same password you use everywhere else!) Change the first few static variables in the code to use your stalker and your IA.

Before starting the stalker, you must first log in using the SL Viewer with both your IA account and your stalker account and position them close to each other. This way, when they log in the stalker will be near the IA and will start following him. Also add each other as friends and set permissions to allow them to see each other's online status in the Contacts List (this will help with debugging). Now log in to your IA account. Once the IA is logged in, start your stalker code. You should see the stalker appear in SL and start following you around. Unfortunately, you cannot have your stalker code running and logged in with the SL Viewer at the same time, so

there is no way to tell if you stalker bot is actually working unless you are logged in as the IA and looking at the stalker.

Part 1 – Follow Using the Database

Currently, the stalker is following the IA by querying the current sim, getting the location of the IA, and moving to that location. We would like the stalker instead to get the location using the database. Recall that the avatar shadow is recording the location of the IA every 10 seconds. In part 1, extend your stalker to do the following:

1. Using a timer, every 10 seconds get the most recent location of the IA from the database. You can use the getPlaces() method from the DB API.
2. If the timestamp last recorded position of the IA is within 20 seconds of the current time, move to that location. Otherwise, do nothing.

If you have implemented part 1 correctly, you will see that the stalker will still follow the IA, but with a delay.

Part 2 – No Escape By Teleport

You will notice that the stalker bot does have one weakness: if the IA teleports to a different region the stalker bot does not teleport after the IA. We want to our stalker to teleport after the IA, so there's no escape! In part 2, extend your code from part 1 to do the following:

1. After getting the most recent location of the IA, check the distance from the stalker's current location.
2. If the stalker is more than 100m away from the IA, teleport to the IA location.
3. Otherwise, if the stalker is within 100m of the IA, continue following the IA as in part 1.

Test your improved stalker by teleporting the IA to different locations in SL and see if the stalker follows you shortly thereafter. *Make sure the IA doesn't end up in a region with scripts disabled. Otherwise, the IA's location won't be get recorded by the avatar shadow and the stalker bot won't be able to follow.*

Part 3 – Repeat What I Say

Is there anything more annoying that someone who follows you around? What if that person also repeated everything you say? In part 3, make the stalker bot even more annoying by repeating anything said by the IA in local chat and append "Echo:" in front of it. For example, if the IA says "You're annoying" the stalker should say "Echo: You're annoying".

The stalker should only repeat what the IA says and no one else's chats. *Hint: upon hearing a chat message, the stalker should check who the speaker was before echoing. You don't want the stalker to end up echoing his own chat messages, which leads to an infinite loop (from firsthand experience, I can say this is not pretty).*

Submission

Please make all your changes on the Follow.cs file provided. Rename your updated file Follow_<sunetid>.cs and the name of the class Follow_<sunetid>. For example, I would rename the file to Follow_hsuray.cs and the class to Follow_hsuray. Email the file to the staff mailing list. I will be testing your code by having it follow my own IA.